

---

Subject: sharing information across widget hierarchies  
Posted by [Chris\[6\]](#) on Sat, 07 Aug 2010 02:23:36 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi everyone (bear with me),

What is the recommended practice for creating multiple widget hierarchies (in separate windows) that share event and state information nicely?

For example, lets say I create two top level bases (2 and 3) which both point to widget 1 as their group leader. I'd LIKE to attach an event handler to widget 1 for simplicity, and store all of the state information in widget 1's uvalue. Unfortunately, I don't think IDL propagates widget events to group leaders. So that won't work.

Ok, I can get around this by attaching the `_same_` event function to widgets 2 and 3. Now at least a single function gets called whenever an event happens; but there's no single widget in which to store state information. So then I have to store the state information as a `_pointer_` in the uvalue to widgets 2 and 3. This means that I'm constantly dereferencing the state information pointer, which is much more awkward (syntactically) than working with a single, non-pointer variable (does anyone else hate how ugly `x = (*state_ptr).big_array_ptr[first_col, *]` looks?)

Am I being too picky? Is there some nice way to create widgets in separate windows that nevertheless behave like a single widget hierarchy as far as events are concerned?

chris

---